**P01\_programavimo elementai**

1. **Kai kuriais atvejais tą patį simbolių spausdinimo rezultatą galima pasiekti nesinaudojant sąlygos sakiniu, bet panaudojant vienas į kitą įdėtus 2 ciklo sakinius (ciklą cikle). Parenkite tokią programą. Ciklų kintamieji turi būti skirtingi. (1.2 skyrius)**

Pats paprasčiausias sprendimas: sukonstruoti stačiakampį, sudarytą iš žvaigždučių. Žvaigždučių skaičius eilutėje yra lygus n, eilučių yra m.

|  |  |
| --- | --- |
| ...  int n, m;  Console.Write("Žvaigždučių skaičius eilutėje: ");  n = int.Parse(Console.ReadLine());  Console.Write("Eilučių skaičius: ");  m = int.Parse(Console.ReadLine());  for (int i = 1; i <= m; i++)  {  for (int j = 1; j <= n; j++)  Console.Write("\*");  Console.WriteLine();  }  ... |  |

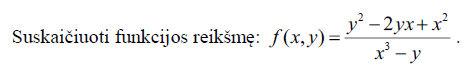
**Įdomesni atvejai:**

Statusis trikampis, sudarytas iš „x“ simbolių. Viršutinėje eilutėje yra vienas simbolis, antroje – du, trečioje – trys ir t.t. Trikampio aukštis – n eilučių.

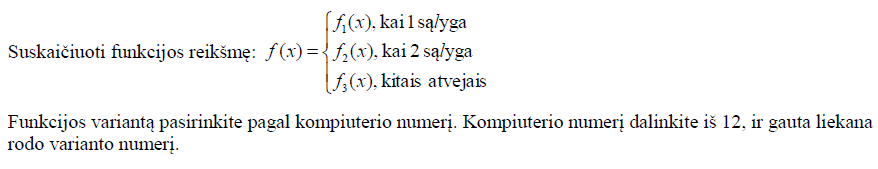
|  |  |
| --- | --- |
| ...  int n;  Console.Write("Trikampio aukštis: ");  n = int.Parse(Console.ReadLine());  for (int i = 1; i <= n; i++)  {  for (int j = 1; j <= i; j++)  Console.Write("x");  Console.WriteLine();  }  ... |  |

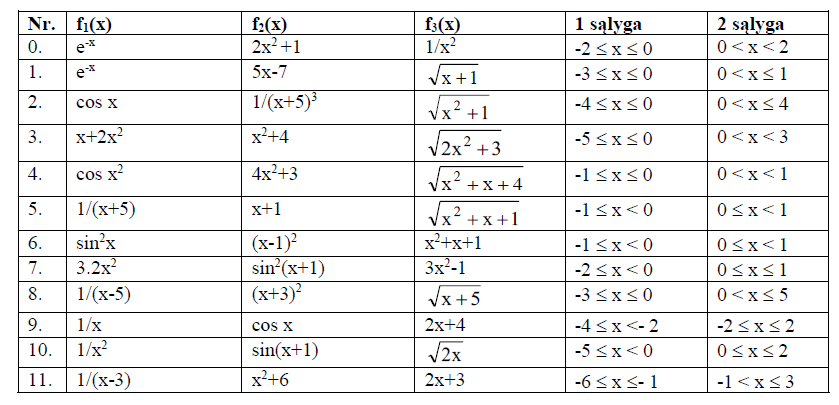
„Apverstas“ statusis trikampis, sudarytas iš „o“ simbolių. Trikampio aukštis – n eilučių. Viršutinėje eilutėje yra n simbolių, antroje – n - 1, trečioje – n - 2 ir t.t.

|  |  |
| --- | --- |
| ...  int n;  Console.Write("Trikampio aukštis: ");  n = int.Parse(Console.ReadLine());  for (int i = n; i >= 1; i--)  {  for (int j = 1; j <= i; j++)  Console.Write("o");  Console.WriteLine();  }  ... |  |

1. ****

|  |  |  |
| --- | --- | --- |
| x | y | f |
| -10 | -10 | -0,22 |
| -9 | -9 | -0,25 |
| -8 | -8 | -0,29 |
| -7 | -7 | -0,33 |
| -6 | -6 | -0,40 |
| -5 | -5 | -0,50 |
| -4 | -4 | -0,67 |
| -3 | -3 | -1,00 |
| -2 | -2 | -2,00 |
| -1 | -1 | neapibrėžta |
| 0 | 0 | neapibrėžta |
| 1 | 1 | neapibrėžta |
| 2 | 2 | 0,67 |
| 3 | 3 | 0,50 |
| 4 | 4 | 0,40 |
| 5 | 5 | 0,33 |
| 6 | 6 | 0,29 |
| 7 | 7 | 0,25 |
| 8 | 8 | 0,22 |
| 9 | 9 | 0,20 |
| 10 | 10 | 0,18 |

1. ****

****

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **x** | **Nr.0** | **Nr. 1** | **Nr.2** | **Nr.3** | **Nr.4** | **Nr.5** | **Nr.6** | **Nr.7** | **Nr.8** | **Nr.9** | **Nr.10** | **Nr.11** |
| -10 | 0,01000 | neapibrėžta | 10,04988 | 14,24781 | 9,69536 | 9,53939 | 91,00000 | 299,00000 | neapibrėžta | -16,00000 | neapibrėžta | -17,00000 |
| -9 | 0,01235 | neapibrėžta | 9,05539 | 12,84523 | 8,71780 | 8,54400 | 73,00000 | 242,00000 | neapibrėžta | -14,00000 | neapibrėžta | -15,00000 |
| -8 | 0,01563 | neapibrėžta | 8,06226 | 11,44552 | 7,74597 | 7,54983 | 57,00000 | 191,00000 | neapibrėžta | -12,00000 | neapibrėžta | -13,00000 |
| -7 | 0,02041 | neapibrėžta | 7,07107 | 10,04988 | 6,78233 | 6,55744 | 43,00000 | 146,00000 | neapibrėžta | -10,00000 | neapibrėžta | -11,00000 |
| -6 | 0,02778 | neapibrėžta | 6,08276 | 8,66025 | 5,83095 | 5,56776 | 31,00000 | 107,00000 | neapibrėžta | -8,00000 | neapibrėžta | -0,11111 |
| -5 | 0,04000 | neapibrėžta | 5,09902 | 45,00000 | 4,89898 | 4,58258 | 21,00000 | 74,00000 | 0,00000 | -6,00000 | 0,04000 | -0,12500 |
| -4 | 0,06250 | neapibrėžta | -0,65364 | 28,00000 | 4,00000 | 3,60555 | 13,00000 | 47,00000 | 1,00000 | -0,25000 | 0,06250 | -0,14286 |
| -3 | 0,11111 | 20,08554 | -0,98999 | 15,00000 | 3,16228 | 2,64575 | 7,00000 | 26,00000 | -0,12500 | -0,33333 | 0,11111 | -0,16667 |
| -2 | 7,38906 | 7,38906 | -0,41615 | 6,00000 | 2,44949 | 1,73205 | 3,00000 | 12,80000 | -0,14286 | -0,41615 | 0,25000 | -0,20000 |
| -1 | 2,71828 | 2,71828 | 0,54030 | 1,00000 | 0,54030 | 0,25000 | 0,70807 | 3,20000 | -0,16667 | 0,54030 | 1,00000 | -0,25000 |
| 0 | 1,00000 | 1,00000 | 1,00000 | 0,00000 | 1,00000 | 1,00000 | 1,00000 | 0,70807 | -0,20000 | 1,00000 | 0,84147 | 6,00000 |
| 1 | 3,00000 | -2,00000 | 0,00463 | 5,00000 | 2,44949 | 1,73205 | 3,00000 | 0,82682 | 16,00000 | 0,54030 | 0,90930 | 8,00000 |
| 2 | 0,13534 | 1,73205 | 0,00292 | 8,00000 | 3,16228 | 2,64575 | 7,00000 | 11,00000 | 25,00000 | -0,41615 | 0,14112 | 14,00000 |
| 3 | 0,04979 | 2,00000 | 0,00195 | 4,58258 | 4,00000 | 3,60555 | 13,00000 | 26,00000 | 36,00000 | 10,00000 | 2,44949 | 24,00000 |
| 4 | 0,01832 | 2,23607 | 0,00137 | 5,91608 | 4,89898 | 4,58258 | 21,00000 | 47,00000 | 49,00000 | 12,00000 | 2,82843 | 11,00000 |
| 5 | 0,00674 | 2,44949 | 5,09902 | 7,28011 | 5,83095 | 5,56776 | 31,00000 | 74,00000 | 64,00000 | 14,00000 | 3,16228 | 13,00000 |
| 6 | 0,00248 | 2,64575 | 6,08276 | 8,66025 | 6,78233 | 6,55744 | 43,00000 | 107,00000 | 3,31662 | 16,00000 | 3,46410 | 15,00000 |
| 7 | 0,00091 | 2,82843 | 7,07107 | 10,04988 | 7,74597 | 7,54983 | 57,00000 | 146,00000 | 3,46410 | 18,00000 | 3,74166 | 17,00000 |
| 8 | 0,00034 | 3,00000 | 8,06226 | 11,44552 | 8,71780 | 8,54400 | 73,00000 | 191,00000 | 3,60555 | 20,00000 | 4,00000 | 19,00000 |
| 9 | 0,00012 | 3,16228 | 9,05539 | 12,84523 | 9,69536 | 9,53939 | 91,00000 | 242,00000 | 3,74166 | 22,00000 | 4,24264 | 21,00000 |
| 10 | 0,00005 | 3,31662 | 10,04988 | 14,24781 | 10,67708 | 10,53565 | 111,00000 | 299,00000 | 3,87298 | 24,00000 | 4,47214 | 23,00000 |

1. **Sukurkite mini skaičiuotuvą darbui su realiais skaičiais. Skaičiuotuvas atlieka 4 operacijas: sudėtį (+), atimtį (-), daugybą (\*), dalybą (/). Operacijas žymėkite atitinkamais simboliais. Įveskite porą skaičių. Įveskite operacijos simbolį. Galima įvesti bet kokį simbolį. Jei įvestas simbolis neatitinka nei vieno operacijos ženklo, turi būti rodomas pranešimas ERROR (arba KLAIDA). Jei simbolis atitinka kurią nors operaciją, suskaičiuokite bei išveskite visus įvestus duomenis ir operacijos rezultatą. Sąlygos sakinyje operacijos ženklą nurodykite tarp apostrofų (viengubų kabučių).**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

1. **Pertvarkykite bent vieną anksčiau sukurtą programą taip, kad skaičiavimai būtų atliekami metoduose.**

Keletas galimų variantų:

2 užduotis. Loginę reikšmę grąžinantis metodas:

static void Main(string[] args)

{

double x, y;

double funkcija;

Console.Write("Įveskite x reikšmę:");

x = double.Parse(Console.ReadLine());

Console.Write("Įveskite y reikšmę:");

y = double.Parse(Console.ReadLine());

if (ArApibrėžta(x, y))

{

funkcija = ((y \* y - 2 \* x \* y + x \* x) / (x \* x \* x - y));

Console.WriteLine(" x = {0}, y = {1}, funkcija = {2,8:f3}", x, y, funkcija);

}

else Console.WriteLine(" x = {0}, y = {1}, funkcija neegzistuoja", x, y);

}

//------------------------------------------------------------

static bool ArApibrėžta(double a, double b)

{

return (Math.Pow(a, 3) - b) != 0;

}

4 užduotis. Du metodai. Vienas tikrina, ar tinkama operacija, antrasis – grąžina apskaičiuotą rezultatą:

static void Main(string[] args)

{

double sk1, sk2;

char operacija;

Console.Write("Įveskite pirmą skaičių: ");

sk1 = double.Parse(Console.ReadLine());

Console.Write("Įveskite antrą skaičių: ");

sk2 = double.Parse(Console.ReadLine());

Console.Write("Įveskite operaciją:");

operacija = char.Parse(Console.ReadLine());

if (Tikrinimas(operacija))

Console.WriteLine(" {0} {1} {2} = {3}", sk1, operacija, sk2,

Rezultatas(sk1, sk2, operacija));

else Console.WriteLine("ERROR");

}

//------------------------------------------------------------------------------------------

static bool Tikrinimas(char op)

{

if (op == '+' || op == '-' || op == '\*' || op == '/') return true;

else return false;

}

//------------------------------------------------------------------------------------------

static double Rezultatas (double sk1, double sk2, char operacija)

{

if (operacija == '+') return sk1 + sk2;

else if (operacija == '-') return sk1 - sk2;

else if (operacija == '\*') return sk1 \* sk2;

else return sk1 / sk2;

}